WEST Search History

DATE: Tuesday, September 23, 2003

Set Name side by side	Query	Hit Count	Set Name result set
DB=JPAB,EPAB,DWPI,TDBD; PLUR=YES; OP=ADJ			
L12	l11 and (second near counter)	2	L12
L11	instruction set or (packet near2 instruction)	3834	L11
DB=USPT; PLUR=YES; OP=ADJ			
L10	L8 and RISC	48	L10
L9	L8 and ROP	0	L9
L8	L7	152	L8
DB=USPT,PGPB; PLUR=YES; OP=ADJ			
L7	L5 and (instruction near (target$ or process$))	164	L7
L6	L5 and (instructionn near (target$ or process$))	0	L6
L5	L4 and (second near counter)	346	L5
L4	instruction set	17883	L4
DB=USPT; PLUR=YES; OP=ADJ			
L3	L2	13	L3
DB=USPT,PGPB; PLUR=YES; OP=ADJ			
L2	L1 and (second near counter)	19	L2
L1	packet near2 instruction	1291	L1

END OF SEARCH HISTORY

1. Document ID: JP 10222367 A US 6055628 A

L12: Entry 1 of 2 File: DWPI Aug 21, 1998

DERWENT-ACC-NO: 1998-511316
DERWENT-WEEK: 199844

TITLE: Data processor with execution delay regulation function - has program counter which sets data reception timing such that second target address is received followed by reception of first target address

INVENTOR: SESHAN, N; SIMAR, L R

PRIORITY-DATA: 1997US-036222P (January 24, 1997), 1998US-0012676 (January 23, 1998)

PATENT-FAMILY:

PUB-NO	PUB-DATE	LANGUAGE	PAGES	MAIN-IPC
JP 10222367 A	August 21, 1998		019	G06F009/38
US 6055628 A	April 25, 2000		000	G06F015/16

INT-CL (IPC): G06 F 9/38; G06 F 15/16

ABSTRACTED-PUB-NO: JP 10222367A
BASIC-ABSTRACT:

The processor (1) has a set of fetching circuits (10a-10c) to perform decoding of the input data. The pipeline execution of each instruction in the input data is regulated by a CPU such that first branch instruction containing first target address is processed for predefined times. A program counter sets reception timing of target address and fetch address corresponding to each phase of pipeline instruction. The fetching of the fetch instruction packet is performed by the fetch circuit based on the designated fetch address. Then, execution of fetch instruction is carried out.

The execution of a second branch instruction corresponds to a second target address is started by a controller (100) after starting the execution of first branch instruction. The reception timing is set by the program counter such that a second target address is received immediately after the reception of first target address.

ADVANTAGE - Eliminates processing delay between pipeline execution. Facilitates execution of instruction without interruption.
ABSTRACTED-PUB-NO:

US 6055628A EQUIVALENT-ABSTRACTS:

The processor (1) has a set of fetching circuits (10a-10c) to perform decoding of the input data. The pipeline execution of each instruction in the input data is regulated by a CPU such that first branch instruction containing first target address is processed for predefined times. A program counter sets reception timing of target address and fetch address corresponding to each phase of pipeline instruction. The fetching of the fetch instruction packet is performed by the fetch circuit based on the designated fetch address. Then, execution of fetch instruction is carried out.

The execution of a second branch instruction corresponds to a second target address is started by a controller (100) after starting the execution of first branch

instruction. The reception timing is set by the program counter such that a second target address is received immediately after the reception of first target address.

ADVANTAGE - Eliminates processing delay between pipeline execution. Facilitates execution of instruction without interruption.

☐ 2. Document ID: JP 08237307 A JP 3189612 B2

L12: Entry 2 of 2 File: DWPI Sep 13, 1996

DERWENT-ACC-NO: 1996-470590
DERWENT-WEEK: 200142

TITLE: Audio encoding/decoding appts used in multi channel radio communication system - uses packet transmitting controller to output start instruction to packet transmission processing part according to value of second counter

PRIORITY-DATA: 1995JP-0036819 (February 24, 1995)

PATENT-FAMILY:

PUB-NO	PUB-DATE	LANGUAGE	PAGES	MAIN-IPC
JP 08237307 A	September 13, 1996		005	H04L012/66
JP 3189612 B2	July 16, 2001		005	H04L012/66

INT-CL (IPC): G10 L 9/18; H03 M 7/30; H04 L 12/56; H04 L 12/66

ABSTRACTED-PUB-NO: JP 08237307A
BASIC-ABSTRACT:

The appts uses a buffer memory (12) to store an audio frame received from radio communication network into PTN through a counter (11) for the number of frames. To operate a frame transmitting controller (14), output of the counter is referred. When predetermined number of frames were stored into the buffer memory, then transmitting instruction is sent to a frame transmitting processing part (13).

When an audio frame is received into the radio communication network from PTN, it is stored into the buffer through a second counter (16). Referring to the value of the second counter, a packet transmitting controller (18) checks out if predetermined number of frames were stored in the buffer. Accordingly, a transmitting instruction is sent to a packet transmission processing part (15).

USE/ADVANTAGE - In Mobile communication system, public telecommunication network. Enables bidirectional data transmission. Prevents clock slippage.

Full | Title | Citation | Front | Review | Classification | Date | Reference | Sequences | Attachments | Claims | KWIC | Draw Desc | Clip Img | Image

---

Generate Collection | Print

Terms	Documents
l11 and (second near counter)	2

Display Format: - Change Format

Search Results

Search Results for: **[instruction <AND>((second counter))]**
Found **14** of **121,259 searched.**

Search within Results

> Advanced Search

> Search Help/Tips

Sort by: Title Publication Publication Date Score Binder

Results 1 - 14 of 14 short listing

1 Reducing cache misses using hardware and software page placement 80%
Timothy Sherwood , Brad Calder , Joel Emer
Proceedings of the 13th international conference on Supercomputing May 1999

2 A logic simulation machine 77%
M. Abramovici , Y. H. Levendel , P. R. Menon
Proceedings of the 9th annual symposium on Computer Architecture April 1982
Special-purpose CAD hardware is increasingly being considered as a means to meet the challenge posed to conventional (software-based) CAD tools by the growing complexity of VLSI circuits. In this paper we describe the architecture of a logic simulation machine employing distributed and parallel processing. Our architecture can accommodate different levels of modeling ranging from simple gates to complex functions, and support timing analysis. We estimate that simulation implemented b ...

3 A type driven hardware engine for Prolog clause retrieval over a large knowledge base 77%
K.-F. Wong , M. H. Williams
ACM SIGARCH Computer Architecture News , Proceedings of the 16th annual international symposium on Computer architecture April 1989
Volume 17 Issue 3
Whereas existing Prolog systems are very effective at handling small knowledge bases, they are not very efficient at and often incapable of handling large sets of clauses. Large knowledge bases which may comprise millions of clauses and are shared by a number of users, may need to reside in secondary memory. In such cases exhaustive search is inordinately slow. Various approaches have been put forward for handling the problem, most of which involve coupled systems (loosely ...

4 High-level debugging in parasight 77%
Ziya Aral , Ilya Gertner

ACM SIGPLAN Notices , Proceedings of the 1988 ACM SIGPLAN and SIGOPS workshop on Parallel and distributed debugging November 1988
Volume 24 Issue 1
Debugging parallel programs with time critical dependencies is difficult due to subtle race conditions that may cause deadlock, starvation, and other errors. These errors can be detected by multiple instrumentation points triggered by logical assertions. Although some advanced debuggers provide a programmer with the ability to define complex logical assertions, they are inadequate for debugging parallel programs due to the high overhead of monitoring these assertions. This paper ...

5 A simple method for extracting models for protocol code 77%
David Lie , Andy Chou , Dawson Engler , David L. Dill
ACM SIGARCH Computer Architecture News , Proceedings of the 28th annual international symposium on Computer architecture May 2001
Volume 29 Issue 2

The use of model checking for validation requires that models of the underlying system be created. Creating such models is both difficult and error prone and as a result, verification is rarely used despite its advantages. In this paper, we present a method for automatically extracting models from low level software implementations. Our method is based on the use of an extensible compiler system, xg++, *to perform the extraction. The extracted model is combined with a model of the ha ...*

6 Proposal for a feasible programming system 77%
Philip R. Bagley
Communications of the ACM August 1959
Volume 2 Issue 8

7 FLOWTRACE, a computer program for flowcharting programs 77%
Philip M. Sherman
Communications of the ACM December 1966
Volume 9 Issue 12
The FLOWTRACE system produces flowcharts of programs written in “almost any” programming language. One must describe the syntax of the control statements in his language; for this purpose a metalanguage is available. The resultant object deck is used to flowchart any programs in the language described. Several examples of FAP and SNOBOL flowcharts are given. However, it is not necessary to confine one's scope to existing languages. One may define his own language in a ...

8 On Formalisms for Turing Machines 77%
Patrick C. Fischer
Journal of the ACM (JACM) October 1965
Volume 12 Issue 4

9 Programmed Multiplication on the IBM 407 77%
Roger L. Boyell
Journal of the ACM (JACM) October 1957

Volume 4 Issue 4

10 Post-mortem black-box correctness tests for basic parallel data structures 77%
Phillip B. Gibbons , John L. Bruno , Steven Phillips
Proceedings of the eleventh annual ACM symposium on Parallel algorithms and architectures June 1999

11 Garbage collecting the Internet: a survey of distributed garbage collection 77%
Saleh E. Abdullahi , Graem A. Ringwood
ACM Computing Surveys (CSUR) September 1998
Volume 30 Issue 3
Internet programming languages such as Java present new challenges to garbage-collection design. The spectrum of garbage-collection schema for linked structures distributed over a network are reviewed here. Distributed garbage collectors are classified first because they evolved from single-address-space collectors. This taxonomy is used as a framework to explore distribution issues: locality of action, communication overhead and indeterministic communication latency.

12 Automated hoarding for mobile computers 77%
Geoffrey H. Kuenning , Gerald J. Popek
ACM SIGOPS Operating Systems Review , Proceedings of the sixteenth ACM symposium on Operating systems principles October 1997
Volume 31 Issue 5

13 Reconciling responsiveness with performance in pure object-oriented languages 77%
Urs Hölzle , David Ungar
ACM Transactions on Programming Languages and Systems (TOPLAS) July 1996
Volume 18 Issue 4
Dynamically dispatched calls often limit the performance of object-oriented programs, since opject-oriented programming encourages factoring code into small, reusable units, thereby increasing the frequency of these expensive operations. Frequent calls not only slow down execution with the dispatch overhead per se, but more importantly they hinder optimization by limiting the range and effectiveness of standard global optimizations. In particular, dynamically dispatched calles prevent stand ...

14 Source level debugging of automatically parallelized code 77%
Robert Cohn
ACM SIGPLAN Notices , Proceedings of the 1991 ACM/ONR workshop on Parallel and distributed debugging December 1991
Volume 26 Issue 12

Results 1 - 14 of 14 short listing

Your search matched **1** of **971569** documents.

A maximum of **1** results are displayed, **25** to a page, sorted by **Relevance** in **descending** order.
You may refine your search by editing the current search expression or entering a new one the text box.
Then click **Search Again**.

('instruction counter')

Search Again

Results:
Journal or Magazine = **JNL** Conference = **CNF** Standard = **STD**

1 **Support for software interrupts in log-based rollback-recovery**
Slye, J.H.; Elnozahy, E.N.;
Computers, IEEE Transactions on , Volume: 47 Issue: 10 , Oct. 1998
Page(s): 1113 -1123

[Abstract] [PDF Full-Text (120 KB)] **IEEE JNL**

Your search matched **1** of **971569** documents.

A maximum of **1** results are displayed, **15** to a page, sorted by **Relevance** in **descending** order.
You may refine your search by editing the current search expression or entering a new one the text box.

Then click **Search Again**.

('second counter')

Search Again

Results:
Journal or Magazine = **JNL** Conference = **CNF** Standard = **STD**

1 **Improvements for the next generation of FDDI**
Game, D.; Maly, K.;
Computers and Communications, 1991. Conference Proceedings., Tenth Annual International Phoenix Conference on , 27-30 March 1991
Page(s): 710 -716

[Abstract] [PDF Full-Text (512 KB)] **IEEE CNF**